CLAIMS

WE CLAIM:

1.   In a computer system, a method of encoding audio data comprising:

encoding a first portion of an audio data sequence in a direct variable-dimension

5   vector Huffman encoding mode;

switching to a run-level encoding mode at a switch point; and

encoding a second portion of the audio data sequence in the run-level encoding

mode.

10   2.   The method of claim 1 further comprising sending a flag in an encoded

bitstream, wherein the flag indicates the switch point.

3.   The method of claim 1 wherein the first portion of the audio data

sequence consists primarily of non-zero quantized audio coefficients, and wherein the

15   second portion of the audio data sequence consists primarily of zero-value quantized

audio coefficients.

4.   The method of claim 1 wherein the switch point is a pre-determined

switch point.

20

5.   The method of claim 4 wherein the pre-determined switch point is

determined experimentally by testing efficiency of encoding the audio data sequence

using the pre-determined switch point.

25   6.   The method of claim 1 wherein the switch point is adaptively

determined.

7. The method of claim 1 further comprising:

switching to a third encoding mode at a second switch point.

8. The method of claim 1 wherein the run-level encoding mode comprises context-based arithmetic encoding of run lengths and levels.

9. The method of claim 1 wherein the run-level encoding mode comprises Huffman coding of run lengths and levels.

10. The method of claim 1 wherein the run-level encoding mode comprises vector Huffman coding of run lengths and levels.

11. A computer-readable medium storing computer-executable instructions for causing an audio encoder to perform the method of claim 1.

12. In a computer system, a method of encoding audio data comprising:

encoding a first portion of an audio data sequence in a direct context-based arithmetic encoding mode;

switching to a run-level encoding mode at a switch-point; and

encoding a second portion of the audio data sequence in the run-level encoding mode.

13. The method of claim 12 wherein the run-level encoding mode comprises context-based arithmetic encoding of run lengths and levels.

14. A computer-readable medium storing computer-executable instructions for causing an audio encoder to perform the method of claim 12.

15.   In a computer system, a method of decoding audio data comprising:

decoding a first portion of an encoded audio data sequence in a direct variable-dimension vector Huffman decoding mode;

switching to a run-level decoding mode at a switch point; and

5   decoding a second portion of the encoded audio data sequence in the run-level decoding mode.

16.   The method of claim 15 further comprising:

prior to the switching, receiving a flag indicating the switch point.

10

17.   The method of claim 15 wherein the first portion of the encoded audio data sequence consists primarily of non-zero quantized audio coefficients, and wherein the second portion of the encoded audio data sequence consists primarily of zero-value quantized audio coefficients.

15

18.   The method of claim 15 wherein the switch point is a pre-determined switch point.

19.   The method of claim 15 wherein the switch point is adaptively

20   determined.

20.   The method of claim 15 further comprising:

switching to a third decoding mode at a second switch point.

25   21.   The method of claim 15 wherein the run-level decoding mode comprises context-based arithmetic decoding of run lengths and levels.

22. The method of claim 15 wherein the run-level decoding mode comprises Huffman decoding of run lengths and levels.

23. The method of claim 15 wherein the run-level decoding mode comprises vector Huffman decoding of run lengths and levels.

24. A computer-readable medium storing computer-executable instructions for causing an audio decoder to perform the method of claim 15.

25. In a computer system, a method of decoding audio data comprising:
decoding a first portion of an encoded audio data sequence in a direct context-based arithmetic decoding mode;
switching to a run-level decoding mode at a switch-point; and
decoding a second portion of the audio data sequence in the run-level decoding mode.

26. The method of claim 25 wherein the run-level decoding mode comprises context-based arithmetic decoding of run lengths and levels.

27. A computer-readable medium storing computer-executable instructions for causing an audio decoder to perform the method of claim 25.

28. In a computer system, a method of encoding data comprising plural symbols, the method comprising:
encoding a first vector comprising a first number of symbols, wherein the encoding the first vector comprises:
selecting a first code table from a set of plural code tables based on the first number of symbols; and

representing the first vector with a code from the first code table; and

encoding a second vector comprising a second number of symbols, wherein the second number of symbols differs from the first number of symbols.

5   29.   The method of claim 28 wherein the first number of symbols is greater than the second number of symbols, and wherein the first vector has a higher probability of occurrence than the second vector.

30.   The method of claim 28 wherein the encoding the second vector

10   comprises:

selecting a second code table based on the second number of symbols; and

representing the second vector with a code from the second code table, wherein the second code table differs from the first code table.

15   31.   The method of claim 28 wherein the first code table comprises:

codes for representing probable vectors of a set of possible vectors having the first number of symbols; and

an escape code.

20   32.   The method of claim 28 wherein the first number of symbols differs from the second number of symbols by a factor of 2.

33.   The method of claim 28 wherein the second number of symbols is 1, and wherein the encoding a second vector comprises:

25   representing the second vector with a code obtained by a table-less encoding technique.

34.   The method of claim 28 further comprising:

encoding a third vector comprising a third number of symbols, wherein the third

number of symbols differs from the first number of symbols and the second number of

symbols, and wherein the encoding the third vector comprises:

selecting a third code table based on the third number of symbols; and

5   representing the third vector with a code from the third code table,

wherein the third code table differs from the first code table and the second code

table.

35.   A computer-readable medium storing computer-executable instructions

10   for causing a computer perform the method of claim 28.

36.   In a computer system, a method of decoding data comprising plural

encoded vectors, the method comprising:

decoding a first vector, wherein the decoding the first vector comprises:

15   receiving a first code representing a first vector, the first vector having a

first number of symbols;

looking up the first code in a first code table in a group of plural code

tables;

determining whether the first code is an escape code;

20   if the first code is an escape code:

receiving a second code representing the first vector, wherein the

second code is not included in the first code table; and

decoding the second code;

if the first code is not an escape code:

25   looking up symbols for the first vector in the first code table; and

including the symbols in a decoded data stream;

wherein the first number of symbols is a basis for whether the first code

is an escape code or not an escape code.

37. The method of claim 36 wherein the decoding the second code comprises:

looking up the second code in a second code table in the group of plural code

5   tables;

determining whether the second code is an escape code;

if the second code is an escape code:

receiving a third code representing the first vector, wherein the third

code is not included in the second code table; and

10   decoding the third code; and

if the second code is not an escape code:

looking up symbols for the first vector in the second code table; and

including the symbols in the decoded data stream;

wherein the second code table differs from the first code table.

15

38. The method of claim 36 wherein the first vector comprises one symbol,

and wherein the decoding the second code comprises:

finding the one symbol using a table-less decoding technique; and

including the one symbol in the decoded data stream.

20

39. The method of claim 36 further comprising:

decoding a second vector, wherein the second vector has a second number of

symbols, and wherein the first number differs from the second number by a factor of 2.

25   40. A computer-readable medium storing computer-executable instructions

for causing a computer perform the method of claim 36.

41.   In a computer system, a method of encoding audio data coefficients using a table-less encoding technique, the method comprising:

if an audio data coefficient is within a first value range, encoding the audio data coefficient with a one-bit code followed by an 8-bit encoded value;

5   if the audio data coefficient is within a second value range, encoding the audio data coefficient with a two-bit code followed by a 16-bit encoded value;

if the audio data coefficient is within a third value range, encoding the audio data coefficient with a three-bit code followed by a 24-bit encoded value; and

if the audio data coefficient is within a fourth value range, encoding the audio

10   data coefficient with a different three-bit code followed by a 31-bit encoded value;

42.   A computer-readable medium storing computer-executable instructions for causing an audio encoder to perform the method of claim 41.

15   43.   In a computer system, a method of encoding audio data in a vector Huffman encoding scheme, the method comprising:

determining a Huffman code from a Huffman code table to use for encoding a vector of audio data symbols, wherein the determining is based on a sum of values of the audio data symbols; and

20   encoding the vector of audio data symbols using the Huffman code.

44.   The method of claim 43 wherein the Huffman code is an escape code, wherein the vector of audio data symbols is an n-dimension vector, and wherein the escape code indicates that the n-dimension vector is to be encoded as x n/x-dimension

25   vectors using at least one different Huffman code table.

45.   The method of claim 43 wherein the determining comprises comparing the sum of values with a threshold, and wherein the threshold depends on a number of audio data symbols in the vector.

5   46.   The method of claim 45 wherein the number of audio data symbols is 4, and wherein the threshold is 6.

47.   The method of claim 45 wherein the number of audio data symbols is 2, and wherein the threshold is 16.

10

48.   The method of claim 45 wherein the number of audio data symbols is 1, and wherein the threshold is 100.

49.   A computer-readable medium storing computer-executable instructions

15   for causing an audio encoder to perform the method of claim 43.

50.   In a computer system, a method of encoding a sequence of audio data coefficients, the method comprising:

receiving the sequence of audio data coefficients; and

20   encoding at least part of the sequence of audio data coefficients using context-based arithmetic encoding.

51.   In a computer system, a method of decoding an encoded sequence of audio data coefficients, the method comprising:

25   receiving the encoded sequence of audio data coefficients; and

decoding at least part of the encoded sequence of audio data coefficients using context-based arithmetic decoding.

52.   In a computer system, a method of encoding audio data coefficients using context-based arithmetic coding, wherein one or more contexts have associated probability distributions, and wherein the probability distributions represent probabilities of coefficients in an audio input sequence, the method comprising:

        adaptively determining a context for a current coefficient in the audio input sequence based at least in part on a mode of representation of the current coefficient; and

        encoding the current coefficient using the context.

53.   The method of claim 52 wherein the mode of representation is direct representation, and wherein the adaptively determining a context for the current coefficient further comprises:

        finding direct levels of previous coefficients in the audio input sequence; and

        determining the context based at least in part on the direct levels of the previous coefficients.

54.   The method of claim 53 wherein the previous coefficients comprise two coefficients immediately preceding the current coefficient in the audio input sequence.

55.   The method of claim 52 wherein the mode of representation is run-level representation, and wherein the adaptively determining a context for the current coefficient further comprises:

        finding a percentage of zero-value coefficients in the audio input sequence;

        finding a previous run length for a previous run of zero-value coefficients in the audio input sequence; and

        determining the context based at least in part on the percentage of zero-value coefficients and the previous run length.

56.   The method of claim 52 wherein the mode of representation is run-level representation, and wherein the adaptively determining a context for the current coefficient further comprises:

finding a current run length of a current run of zero-value coefficients in the audio input sequence;

finding a previous run length of a previous run of zero-value coefficients in the audio input sequence;

finding previous direct levels of previous coefficients in the audio input sequence; and

determining the context based at least in part on the current run length, the previous run length, and the previous direct levels.

57.   A computer-readable medium storing computer-executable instructions for causing a computer to perform the method of claim 52.

58.   In a computer system, a method of encoding audio data using an entropy encoder, the method comprising:

coding a first portion of the audio data, wherein the coding of the first portion comprises direct coding of coefficients;

maintaining a count of consecutive coefficients equal to a predominant value, wherein the count increases if a current coefficient's value is equal to the predominant value; and

in response to the count exceeding a threshold, coding a second portion of the audio data, wherein the coding of the second portion comprises run-level coding of coefficients.

59.   The method of claim 58 wherein the predominant value is zero.

60.   The method of claim 58 wherein the threshold value is static.

61.   The method of claim 58 wherein the threshold is determined adaptively.

5   62.   The method of claim 58 wherein the first portion comprises a block of coefficients, the block having a size, and wherein the threshold depends on the size.

63.   The method of claim 62 wherein the size is 256 coefficients, and wherein the threshold is 4.

10

64.   The method of claim 62 wherein the size is 512 coefficients, and wherein the threshold is 8.

65.   In a computer system, a method of decoding audio data using an entropy

15   decoder, the method comprising:

decoding a first portion of the audio data, wherein the coding of the first portion comprises direct decoding of coefficients;

maintaining a count of consecutive coefficients equal to a predominant value, wherein the count increases if a current coefficient's value is equal to the predominant

20   value; and

in response to the count exceeding a threshold, decoding a second portion of the audio data, wherein the decoding of the second portion comprises run-level decoding of coefficients.

25   66.   The method of claim 65 wherein the predominant value is zero.

67.   The method of claim 65 wherein the threshold is static.

68. The method of claim 65 wherein the threshold is determined adaptively.

69. The method of claim 65 wherein the first portion comprises a block of coefficients, the block having a size, and wherein the threshold depends on the size.

5

70. The method of claim 69 wherein the size is 256 coefficients, and wherein the threshold is 4.

71. The method of claim 69 wherein the size is 512 coefficients, and wherein

10 the threshold is 8.

72. In a computer system, a method of encoding audio data comprising a sequence of audio coefficients using run-level encoding, the method comprising:

encoding a first portion of the sequence using codes from a first code table,

15 wherein the first code table is used by an encoder when longer runs of consecutive coefficients equal to a predominant value are more likely; and

encoding a second portion of the sequence using codes from a second code table, wherein the second code table is used by the encoder when shorter runs of consecutive coefficients equal to the predominant value are more likely.

20

73. The method of claim 72 wherein the predominant value is zero.

74. The method of claim 72 further comprising indicating use of the first or second code table with a signal bit.

25

75. In a computer system, a method of entropy decoding audio data comprising a sequence of audio coefficients using run-level decoding, the method comprising:

decoding a first portion of the sequence by looking up codes in a first code table, wherein the first code table is used by a decoder when longer runs of consecutive coefficients equal to a predominant value are more likely; and

decoding a second portion of the sequence by looking up codes in a second code table, wherein the second code table is used by the decoder when shorter runs of consecutive coefficients equal to a predominant value are more likely.

76.   The method of claim 75 wherein the predominant value is zero.

77.   The method of claim 75 further comprising, prior to the decoding the first portion:

receiving a signal bit indicating whether the first or second code table was used by an encoder.